Twixtor™ User's Manual.

NOTE: **We have tutorial projects for common Twixtor uses.** See our website, http://www.revisionfx.com. Go to the "Twixtor" section and navigate to the help page. You will find tutorial projects for

- After Effects 5.0 and later (windows), 7.0 and later (macintosh)
- combustion 3.0.4 and later
- Premiere Pro CS2 and later

What is Twixtor?

Twixtor intelligently slows down, speeds up or changes the frame rate of your image sequences.

Twixtor enables you to "time warp" your image sequences with visually stunning results. In order to achieve its unparalleled image quality, **Twixtor** synthesizes unique new frames by warping and interpolating frames of the original sequence... employing RE:Vision's proprietary tracking technology that calculates motion for each individual pixel.

Twixtor works in Adobe (R) **After Effects (R),** Adobe **Premiere Pro (R), Autodesk Combustion (TM),** and **other programs that support After Effects plugins. See website for exact compatibility.**

Twixtor Features:

- Calculation of motion vectors at each pixel
- Warping and interpolation of frames to "time warp" input footage
- Smart processing of fields in the input and output as appropriate
- Stretching of a sequence using a single scaling parameter
- Keyframeable retiming for complete control on a frame-by-frame basis.
- Frame rate conversions made easy. Frame rate conversions only supported in After Effects and combustion
- Ability to mark cut points so that material with edits can be retimed or transcoded without breaking the material into pieces first
- With Twixtor 4 and within After Effects, you can use the AE's time remap to speed up, slow down and time remap tools to control the timing of both audio and imagery and use **Twixtor** "underneath" to control the process of time warping the imagery.
- 8 and 16 bits per channel support. Floating point support in After Effects 7.0 or later.

Twixtor Pro adds many features that help you successfully retime clips with objects that pass in front of each other. **Twixtor Pro** allows you to guide **Twixtor**'s tracking on problematic footage. In particular **Twixtor Pro** adds the following features:

- The ability to separate the source material into multiple layers that are inbetweened individually. Layers in **Twixtor Pro** are specified through the use of mattes that you create (independently of **Twixtor**). Separate layers are particularly useful when objects in the scene cross each other. With **Twixtor Pro**, up to 4 separate layers can be specified.

- Up to 12 tracking points can be specified to help guide **Twixtor**'s motion estimation. By using the

tracking points you can explicitly tell Twixtor where a pixel moves from one frame to the next in order to guide **Twixtor**'s calculation of motion vectors. You can set the position of each point at each frame by hand, but more importantly, these points can be positioned from frame-to-frame using the host application's point-tracking features, when available. (Please read your host application manual for availability and instructions for setting plugin position settings via point tracking)

- **Twixtor Pro** provides the option to output the per-pixel motion vectors that are calculated (this plugin is only for applications that support 16 bpc and floating point processing and output)
- **Twixtor Pro** can remap time using motion vectors supplied. Motion vectors can come from many places, such as a 3D animation system or from the **Twixtor Pro** motion vector output plugin (only recommended for applications that support 16 bpc and floating point processing and output).

For After Effects 5.0 (and later) and combustion 3.0 (and later) only:

- **Twixtor Pro**'s tracking can be interactively guided and corrected using the host application's splines and polylines! When **Twixtor** exhibits tracking problems, you can guide Twixtor by simply creating and animating splines to show **Twixtor** where objects are actually moving. Furthermore, the guiding shapes need not be closed.

Before You Start, EXTREMELY IMPORTANT.
Common problems discussed here!

For all applications

- **NEW FOR Twixtor 4.** **Twixtor** now respects the field rendering settings of the host application. **Make sure to turn Field Rendering on in the final render if you want interlaced output**! Pre-**Twixtor** 4 users take note: because **Twixtor** 4 respects the field rendering setting of the host application, there is no longer an Output:Fields setting in **Twixtor**.

  PLEASE READ: The document "How to set a host application's fields' settings for RE:Vision Effects products that work directly with fields" describes what you need to do in the host application in order to use **Twixtor**.

- We will often refer to footage as "30p" or "30i." By that we mean a frame rate of "30 frames per second" and either "progressive" (the p designator) or "interlaced" (the i designator). In addition, we often refer to NTSC material with a 29.97 FPS frame rate by referring to it as if was 30.

- Another note on fields... if you have 3:2 pulldown in your source footage, you will want to **remove 3:2 pulldown** before processing with **Twixtor**. Duplicate fields will cause **Twixtor** to create incorrect inbetween frames. (If your host application does not supply a 3:2 pulldown removal option, then you probably want to purchase our **FieldsKit** product that has a plugin that does just this).

- Each application names it's grouping and parenting mechanisms something different (in AE they are called "compositions, in Final Cut Pro they are called "sequences"). In this documentation any footage, composition, sequence, or anything else that can be filtered will be called **sources** or **inputs**. **Twixtor** filtered sources will be referred to as **output**.

- When applying **Twixtor** to a source, if the output duration is less than the source duration, then **Twixtor** can be applied to the source directly.

- **If the output is to be made longer than the source,** then some accommodations need to be made!

  PLEASE READ: The document "How to setup a RE:Vision Effects plugin that makes a sequence longer." describes the art of dealing with sequences whose outputs are longer than inputs.

- We allow you to specify which source is to be processed (which may not necessarily be the same as the source to which **Twixtor** is directly applied), via the Color Source parameter (i.e., the source that is processed). If the clip that **Twixtor** is directly applied to, and the source specified in Color Source are not of the same pixel dimensions, then a **semi-transparent red image** will be produced to let you know that there is an error. We provide the ability to specify the input source in the user-interface so that pre-comping, nesting, or parenting can often be avoided. For systems that don't automatically set the Color Source, **Twixtor** will output a **semi-transparent blue image** until the Color Source is set.

- Memory requirements: Let's say your images are of resolution width x height. You'll need 3 image buffers worth of memory (where an image buffer has size 4*width*height bytes while working in 8 bits per channel and 8*width*height bytes for 16 bits per channel) to calculate frames. In addition, if you are using **Twixtor Pro** with foreground mattes (for layer separation), you will need 5 additional images worth of memory per layer.

- Special considerations for particular host applications

  After Effects

  - **NEW for Twixtor 4.** Set the field interpretation in After Effects for your footage as is appropriate. However you also need to set the fields interpretation in the Twixtor plugin. Please read the document "How to set a host application's fields' settings for RE:Vision Effects products that work directly with fields" (for pre-Twixtor 4 users: **Twixtor** no longer needs you to set the Interpret Footage->Fields Separation to "None.")
  - **NEW for Twixtor 4. Twixtor** now respects the field rendering settings of After Effects. **Make sure to turn Field Rendering on in the final render if you want interlaced output**! (for pre-Twixtor 4 users: **Twixtor 4** no longer has an "Output:Fields" setting).
  - As said previously, **Twixtor** by itself cannot make the source it is applied to longer... you'll need to help After Effects. See the following document: "How to setup a RE:Vision Effects plugin that makes a sequence longer."
  - **Twixtor** and "Interpret Footage." If you wish to interpret a piece of footage at a different frame rate, make sure that **Twixtor**'s input frame rate is set to the same value you are using for "interpret footage."

  Combustion

  - **NEW for Twixtor 4.** Set the Fields setting in combustion for your footage as is appropriate. However you also need to set the fields interpretation in the Twixtor plugin. Please read the document "How to set a host application's fields' settings for RE:Vision Effects products that work directly with fields" (for pre-Twixtor 4 users: **Twixtor** no longer needs you to set the Fields setting to "None.")
  - **NEW for Twixtor 4. Twixtor** now respects the field rendering settings of combustion. **Make sure to turn Field Rendering on in the final render if you want interlaced output**! (for pre-Twixtor 4 users: **Twixtor 4** no longer has an "Output:Fields" setting).
  - As said previously, **Twixtor** by itself cannot make the source it is applied to longer... you'll

need to help Combustion. See the following document: "How to setup a RE:Vision Effects plugin that makes a sequence longer."

Premiere Pro CS2 and later

For any Premiere projects using **Twixtor**:

- It is **required** that "Project Settings > General > Playback Settings > Deinterlace Desktop Playback" be NOT checked
- It is suggested that "Project Settings > General > Playback Settings > High-Quality Playback" should be checked.
- It is suggested that "Project Settings > General > Playback Settings > High-Quality Scrubbing" should be checked.
- It is suggested that **Twixtor**'s Color Source be set to Current (which is the default).

- As said previously, **Twixtor** by itself cannot make the source it is applied to longer... you'll need to help Premiere. See the following document: "How to setup a RE:Vision Effects plugin that makes a sequence longer."
- **NEW FOR Twixtor 4. Twixtor 4** no longer needs special settings in Premiere for field handling. However you'll need to set the fields setting for the input footage in the Twixtor plugin. Please read the document "How to set a host application's fields' settings for RE:Vision Effects products that work directly with fields"
- **NEW FOR Twixtor 4. Twixtor 4** can now be used in Premiere interlaced projects. **Twixtor** now respects the field settings of Premiere project: For Premiere Pro interlaced projects, Twixtor will produce interlaced material. For Premiere Pro progressive projects, Twixtor produces progressive material. (for pre-Twixtor 4 users: as such, **Twixtor 4** no longer has an "Output:Fields" setting).
- Pre CS2 users will need to use Twixtor 4 or earlier.
- Additional Premiere Support FAQ here: Common Premiere problems

Note: **Twixtor** actually changes the frames that are displayed. However, when you walk along the timeline in Premiere, frames are generally displayed pre-**Twixtor**. To see the Twixtored version, you'll need to Render Scrub (see the Premiere manual), or better yet, Preview the working area to RAM or disk (again, see your Premiere manual).

Other Applications

Commotion and FCP are no longer support with this new AE-compatible version of Twixtor.

Usage of Twixtor

Display:

1. Twixtored Output: This setting displays the final **Twixtor** result. This setting has all the layers that you have specified (via mattes) retimed individually and composited back together.
2. Source: This setting displays the original color source; no retiming is done. This setting is particularly useful when setting **Twixtor**'s Src: Mark Segments setting.

Source Control

Color Source: Color Source specifies the clip to be retimed, or "Twixtored."

> **Color Source must have some value other than "None!"** The Color Source clip is tracked and its color information is used to create new frames. Note that another clip can be specified with Alt Motion Source for tracking purposes, limiting the role of Color Source to be the RGB values that are warped. Until you get familiar with **Twixtor**, simply leave Alt Motion Source set to "None."

Alt Motion Source: This setting is described later in this document, and is designed to help you provide an alternate clip to use for tracking (particularly useful when the source clip is dark or does not have enough contrast for adequate tracking).

Specifying Input: Fields and Input: Frame Rate.

> What are fields?
> 1) Simply put, an interlaced video frame consists of 2 fields. One field consists of all even lines of a frame, and the other field consists of odd lines. A video device (monitor, TV, etc) has a beam that traces, or draws, one field, then goes back and draws the other. As such, one field is displayed first, then the other is displayed half of a frame time later.
>
> 2) So interlaced video has two fields per frame. One field is displayed first in time, and then the other. If the top scan line is part of the field that is displayed first (that is, the field of odd scan lines is displayed first) then we say that the video is "Upper Field First." If the second-to-top scan line is part of the field that is displayed first (that is, the field of even scan lines is displayed first) then we say that the clip is "Lower Field First ".
>
> The rule of thumb when specifying "field first" for your footage: if you choose lower or upper field first and the plugin produces material that is overly jittery, or jumps back and forth, then you probably chose the incorrect option.
>
> Note that you cannot assume all of your source material will have the same field order.
>
> Unfortunately, even though you've specified the field order and frame rate for your footage in the footage settings of your application, **Twixtor** must ask you for this information again because it is not available to our filter from inside the application.
>
> - **Input: Fields**: Specifies whether the input source has no fields, or whether the

upper field or lower field of each frame should be displayed first in time.

- Input: Frame Rate: (For After Effects and combustion only). Specify the input frame rate of your source material.

  Note: If you place your source footage in a 59.94 fps (NTSC) or 50 fps (PAL) composition so that you can view each field as a full frame (or so that you can apply **FieldsKit** to deinterlace each field before applying **Twixtor**), then when you apply **Twixtor** make sure to set Input:Fields to None and the Input:Frame Rate to 59.94 or 50. It's only when you pass Twixtor interlaced material at 29.97 or 25 fps that you should set the Input:Fields to Lower or Upper.

Src: Mark Segments: This setting is described later in this document and is designed to provide an override mechanism for the Motion Vectors calculation based on source material knowledge. Since you need to know what the Motion Vectors setting does, we'll describe the Src: Mark Segments later in this document.

Track Control

Motion Vectors: A menu option with 5 choices for determining the quality of the motion vector calculation.

1. No Motion Vectors: no motion vectors are calculated
2. Sloppy: motion vectors are calculated at 1/4 resolution. Furthermore, a faster but less accurate method is used.
3. Medium: motion vectors are calculated at 1/4 resolution.
4. High: motion vectors are calculated at 1/2 resolution.
5. Best: motion vectors are calculated at full resolution.

   When in "Draft " mode for a host application we calculate motion vectors at half the resolution as specified above.

Important: How output frames are calculated. For demonstration purposes, let's assume that we are slowing down a clip by a factor of 3 (Speed% set to 33.3333). As such, output frame 25 would conceptually need input frame 8.33333. How is frame 8.333 calculated?

- Case 1: Frame Interp: **Nearest** with Motion Vector Quality: **No Motion Vectors**. In this case, no warping can occur (there are no motion vectors!), so the closest input frame in time (input frame 8) is returned. This is equivalent to Time Stretching or Time Remapping in After Effects without frame blending.
- Case 2: Frame Interp: **Blend** or **Motion Weighted Blend** with Motion Vector Quality: **No Motion Vectors**. In this case, no warping can occur (there are no motion vectors!). However, we've asked the module to blend frames, so we'll get a return frame that is a dissolve of frame 8 (at 67%) and frame 9 (33%). This is similar to After Effects with frame blending turned on.
- Case 3: Frame Interp: **Nearest** with Motion Vector Quality set to **anything other than No Motion Vectors**. The motion is calculated from input frame 8 to input frame 9. Frame 8 is then warped using the motion calculated (scaled by 1/3) and returned.
- Case 4: Frame Interp: **Blend** or **Motion Weighted Blend** with Motion Vector Quality set to

anything other than No Motion Vectors. The motion is calculated from input frame 8 to input frame 9. Frame 8 is then warped using the motion calculated (scaled by 1/3) and returned. Frame 9 is warped using the motion calculated from frame 9 to frame 8 (scaled by 2/3). Finally, these two warped frames are blended as in Case 2.

Note that if fewer frames are being generated for the output sequence, that is, the input sequence is being speed up (total time duration is shrunk) OR a lower output frame rate than input frame rate is specified, then motion blur may then be applied (see Motion Blur option below).

Image Prep: A menu option with 2 choices

1. None
2. Contrast/Edge Enhance

The default image preparation is to do nothing special with the source material that is used for tracking. If your material is dark or has poorly defined edges then **Twixtor** might perform better if you choose Contrast/Edge Enhance. Note that the Contrast/Edge Enhance setting may perform more poorly than using the None setting on material that already has reasonable contrast or has many thin edges close together.

Motion Sensitivity: will be explained later in this document.

Plenty Memory? (Cache Last Motion?):

I'm slowing footage down, do I really need to recalculate the motion vectors every frame? The answer is no!

If you have plenty of memory, check the Plenty Memory? (Cache Last Motion?) checkbox. This will cache the last motion vector calculation. This will only be helpful when slowing down footage (and is ignored when speeding up footage). Let's say your image sequence has resolution (width x height). When this option is turned on, you'll need an extra 8*width*height bytes of memory. In addition, if you are using **Twixtor Pro** with foreground mattes (for layer separation), you will an extra 8*width*height bytes worth of memory for each foreground matte you supply.

Warning: If after you've checked Plenty Memory? you change the clip used as the Color Source or Alt Motion Source or FG Mask, please make sure to turn off, then turn on this option in order that this option will function properly (by turning the option off and on we make sure to clear to motion buffers based on the new clip). None of the applications we plug into notify **Twixtor** when you change your clip selections.

Output Control

Changing sequence timing.

Okay, now here's the meat of the plugin. How do I retime my footage?

First let's assume that the input frame rate is the same as the output frame rate.

Time Remap Mode: this menu has 2 options for the retiming method, as follows:

- **Speed**

  When Speed is chosen as the Time Remap Mode, then you can set (and animate) the speed of the input clip. To obtain the speed of the clip, **Twixtor** gets the value from the Speed% slider. A value of 50 (percent) tells **Twixtor** to play the clip 1/2 as fast as the original, a value of 300 tells **Twixtor** to play the original clip at 3 times its normal rate.

  You may ask "*I know that I want to change the duration of the clip by a certain amount, how do I figure out how to set the Speed setting?*"

  In general, if you have a constant factor for a duration change, set Speed% to: 100 / (duration change)

  - For 2X duration change, set Speed% = 100/2 = 50%
  - For a 0.5 duration change (speed up), set Speed% = 100/0.5 = 200%

  Valid Speed% settings are from -10,000% to +10,000 (that is -100 to +100 times original rate). However, the initial slider display in most host applications will be -1000% to +1000% (-10 to +10 original rate). Note that you can type in a higher value even though the slider shows a smaller range than the valid range.

  Note on negative speed: A Speed% of -100.00 does not simply flip the original sequence so that it plays backwards starting at frame 0. Imagine the play head on your clip. If you are at the beginning of the clip and you tell the machine (**Twixtor** in this case) to start playing backward, then there's nothing before the clip starting position to play. That's why you will not see a result with a negative speed as the first keyframe. If you play the clip at 100% for a while, then put in a –30% keyframe (or other negative value) at the place where you want to start playing backward you will see proper results.

- **Frame Num**

  If Frame Num is chosen, then **Twixtor** knows you'll be keyframing your time resequencing by exact frame numbers. **Twixtor** looks at the value of the Frame Num slider to figure out what frame from the input sequence to compute, even if it's a fractional value.

  This section describes keyframing the remapping of time using frames. Again, let's assume that the input frame rate is the same as the output frame rate.

  If you wish to keyframe the timing using direct frame numbers specification, then make sure Time Remap Mode is set to Frame Num. The Speed% setting is ignored in this case, and the Frame Num parameter is used instead. The value of the Frame Num parameter tells **Twixtor** which input frame should be returned for the current time (fractional values are interpolated as described in the previous section).

To use the Frame parameter for retiming:

Immediately after applying **Twixtor** to a clip, it is advisable that you:

First, make the Frame Num parameter animatable.

Second, you are advised to make keyframes for the Frame Num parameter as follows:

- make a keyframe with value 0 at time 0
- make a keyframe at the last time value in the sequence, with a value equal to the last frame number

For example: Let's say you have a sequence of 300 frames.

- make a keyframe with a value of 0 at time 0
- make a keyframe with value 300 at time 300 (time, in this case, is in "frames" instead of timecode).

This will give you an identity mapping (frame 0 returned at time 0, frame 300 returned at time 300, frame 47 will be returned at time 47, etc.).

If Time Remap Mode is set to Speed% then the Frame Num parameter is ignored. Conversely, when the Time Remap Mode is set to Frame Num, the Speed % slider is ignored when key-framing destination frames." TIP: If you are doing a complex dynamic remapping of the video action, you can quickly set your key-frames by setting Motion Vector Quality to None and Frame Interp to Blend. When done setting keyframes, you can go back and set the Motion Vector Quality to the desired quality.

Note for users of After Effects, Combustion
Different Frame Rates

Okay, but what if the input and output frame rates are different? If you don't want to change the duration of the clip, **(and you are using After Effects or Combustion)** you're in luck. Simply set the input and output field rates as appropriate and you are done! The default settings of **Twixtor** will do the right thing for you. (After Effects users take note: you do not set the output frame rate from within **Twixtor** because **Twixtor** can automatically determine the output frame rate of the composition).

If you want to change the duration of the input source at a constant rate, make sure Time Remap Mode is set to Speed and set the Speed% setting the desired amount. **Twixtor** automatically accounts for the differences in frame rates. The Speed mode will vary the total number of seconds, and the total number of seconds remains the same regardless of what the output frame rate is. Then, after calculating the change in seconds, **Twixtor** samples the output at the output frame rate.

If you want to convert 1 second of 24fps footage into 2 seconds of 30 fps footage, set Speed% to 50. Again, the Speed% parameter scales the total number of seconds of a clip (not the absolute total number of frames). In order to get a handle on this, think of

Speed% as changing the total number of seconds and not the frame count. (In our case here 1 second of 24fps footage, or 24 frames, is converted to 2 seconds of 30 fps footage, or 60 frames).

Note for users of After Effects, Combustion
Remapping Time with Different Frame Rates

What if you want to dynamically remap time AND change frame rates at the same time?

It's a little tricky to keyframe time on top of a frame rate conversion but it can be done. The keyframing should be done presuming that the input has already been converted to the output frame. To make this concrete: say you have a value of 60 for Frame, and the output frame rate is 30fps and the input frame rate is 24fps. What input frame is retrieved? We first calculate the *time value* of the frame in the output frame rate (frame 60 = 60/30fps the frame at 2.0 seconds). Then we convert this 2.0 seconds time value to the input frame by multiplying by the input frame rate (2.0*24fps = input frame 48). Of course fractional frames are retrieved, warped and blended as described before.

Output characteristics

- Output: Frame Rate: (combustion only). Allows you to specify the output frame rate. We need you to specify this because **Twixtor** cannot retrieve the output frame rate from the composite.
  (with all the other applications **Twixtor** retrieves the output frame rate from the application).
- Output: Fields: If this setting is displayed, set the appropriate output field setting desired. In most applications there is no separate Output:FIelds setting in Twixtor (unlike pre-Twixtor 4 versions). When Output:Fields is not present, **Twixtor** respects the output fields setting of the host application. In After Effects, Boris Red and combustion, interlaced material is only output when field rendering is set appropriately and you are rendering to disk. In Premiere Pro, Twixtor respects the projects settings for producing interlaced material.
  Note that in After Effects and Combustion that you can render each field and view them *interactively* if you put the source material into a 59.94 composition (and of course setting the input frame rate as appropriate and, for combustion, set the Output frame rate to 59.94).

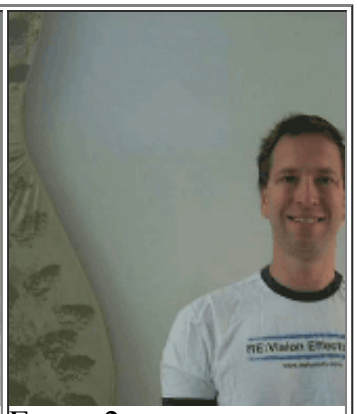
Frame Interp: A menu option with 3 choices.

1. Nearest: warps the nearest calculated input frame by the right amount to make the current output frame.
2. Blend: Performs bi-directional warping and then blends the resulting two warped images. The more you stretch a clip, the more this choice is recommended.
3. Motion Weighted Blend: When a new frame is created from existing frames it is often the case that some softening is introduced. This softening can be more or less strong (depending on how far away in time the new frame is located from existing frames in the original

sequence. This can result in a pulsing of what appears to be the Twixtored sequence coming in and out of focus. The Motion Weighted Blend option attempts to reduce this pulsing.

Smart Blend

Smart Blend gives better results on a pan or zoom or other footage where objects (or parts of objects) are entering or exiting the frame. When this option is off on such shots, you might get pixels that smear at the edge of the frame. When turned on, this option can produce much better blending when objects are moving off screen. This option will increase the calculation time and is only necessary on certain shots (like shots with pans and zooms with greater than 5%), so that's why it's an option.

Frame 1	 Frame 2
Slow Mo 5x without Smart Blend. Note streaking on right edge of **Twixtor** created frames.	Slow Mo 5x with Smart Blend.

Motion Blur Compensation:

When speeding up a clip, you are essentially telling **Twixtor** that there is "more action" happening per frame. In order to simulate what happens in a real camera, and to reduce flickering, **Twixtor** applies motion blur when speeding up footage in order to simulate what happens when more motion takes place in the same amount of (virtual) shutter opening of a camera. You can dial in how much motion blur is added with the Motion Blur Compensation slider. **New in Twixtor 4**: Twixtor also compensates for motion blur by removing motion blur when slowing down footage. Note that the removal of motion blur may be more or less effective depending on how much blur there is in the original sequence.

The default setting for Motion Blur Compensation is 0.0 (that is, by default there is not motion blur compensation). A value of 0.5 is a good place to start if you wish **Twixtor** to automatically compensate for motion blur (this assumes a shutter angle of 180 degrees).

Note that the application of motion blur in **Twixtor** only occurs when footage is sped up or slowed down, to compensate for different virtual shutter speeds. This is different from our ReelSmart Motion Blur product, which adds motion blur based on the motion calculated in a sequence without speeding up or slowing down the footage.

Note for After Effects and combustion users: if you are field rendering on output but viewing only progressive frames interactively (at either 29.97 or 25 fps for NTSC and PAL respectively) please note the following: the resulting output fields will have 1/2 as much motion blur (upon speedup) or twice as sharp (upon slowdown) as when viewing the frames interactively because the interlaced material will have twice as many time samples (one for each field instead of one for each frame). Of course you can always work in a 59.94 or 50 fps comp in order to view each *field* and render out to disk at 29.97 (or 25) fps with field rendering turned in the host application.

When I retime the footage I'd like Twixtor to respect information I know about the source material. How do I do that?

When you retime the footage you may notice that **Twixtor** inappropriately morphs across a hard cut edit in the source material. Or there may be a section of the footage, such as a dissolve, where you know frame blending without any motion vectors would produce better results. Because of these situations we have provided the following setting.

Src: Mark Segments: This setting allows you to specify information you know about the source material. When animating this option, make sure to do it when Display is set to Source. Special note to After Effects, Premiere Pro and combustion users: to remind you to animate this option in the Source timing this option will be unsettable unless Display is set to Source.

After Effects, combustion and Premiere Pro users will see a menu with the following options.

- Cut A
- Cut B
- Cut C

  Twixtor will not interpolate between two segments marked with different Cut settings. For example, if one frame is marked as part of Cut A and the next frame is marked as part of Cut B, then Twixtor will not interpolate between the two when time remapping. In fact, let's say frame 10 is part of Cut A and frame 11 is part of Cut B. Let's say **Twixtor** is asked to slow the footage down: if **Twixtor** is asked to make frame 10.4, then instead of taking 60% of frame 10 and 40% of frame 11, Twixtor will simply return frame 10. If **Twixtor** is asked to make frame 10.6, then frame 11 is simply returned. In fact, when two adjacent frames are marked as part of different cuts, Twixtor simply returns the nearest frame in the source instead of blending between the two frames. However, when two source frames to be interpolated are marked with the same Cut setting, then the Motion Vectors and Frame Interp settings will be used to interpolate between the two frames to create the output frame.

- No MV, Use Nearest
- No MV, Frame Blend

  When the first source frame (of a pair of frames to be interpolated) is marked with either of these two settings, then any output **Twixtor**ed frame made will have the motion vector calculation turned off and the nearest frame (or frame blended version, as selected) will be returned. Think of this setting as overriding the Motion Vectors and Frame Interp settings... but unlike the other two settings which are specified in the output **Twixtor**ed timing, this is specified in the source material's timing. These two settings are useful for a dissolve, wipe or other transition in the source footage will "obviously" produce unwanted artifacts when interpolated using **Twixtor**'s motion vector estimation. In essence this setting lets you turn of motion vectors in the source timing without having to find out where the **Twixtor**ed frames end up in the output timing.

The tracking is not working so well on my source footage, what can I do?

Tracking problems show up as motion not being properly tracked: either objects in the output sequence "gloop" together, or objects don't move enough causing what seems to be simple cross dissolves of the input images. (for what constitutes problems for our tracker, see the "PROBLEMS IN INPUT SEQUENCES" section below.

Alt Motion Source:
You might say, "if only I could enhance the colors used in the tracking process, but then use the colors from the original sequence for the Twixtored sequence." Or, the reverse, you've heavily filtered the original (which you've used as the color source for

Twixtor), but you think you might get better tracking results by using the original footage as the imagery to be tracked. So, we've provided this parameter to set an Alternate Motion Source: If Alt Motion Source is set to None then the Color Source is used for both tracking and the color of the **Twixtor** sequence. However, if an Alt Motion Source clip is specified, then this sequence is used for tracking and the Color Source clip is used for color. Of course, **Twixtor** assumes that both the Alt Motion Source and Color Source clips line up exactly in time.

If an alternate motion clip is supplied, then it must have the same pixel dimensions as the color source; otherwise a **semi-transparent red image** will be produced to let you know that there is an error.

So, how can you enhance tracking using an Alt Motion Source? The following set of pictures seeks to explain:

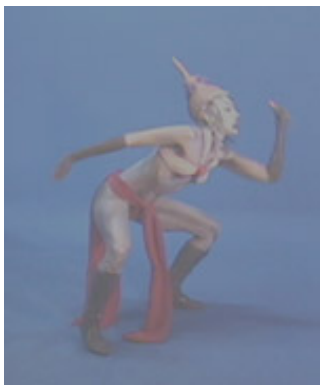


Figure 1

Figure 2

Figure 3

Figure 4

Using an alternate motion source:

- Figure 1: Original low contrast sequence
- Figure 2: Inbetween frame, note that the rear arm is not tracked well because the arm is similar in intensity to the background (of course our eyes see the difference!)
- Figure 3: Image enhanced, but used only for the tracking portion of **Twixtor** by specifying it as the Alt Motion Source.
- Figure 4: The original colors in Figure 1 are used, but the tracking is performed on a sequence represented by Figure 3. Note that the rear arm is now tracked

nicely!

Motion Sensitivity:
Okay, you might say "I don't have time to experiment with image processing my footage to specify an alternate motion source," or worse, "I try to color correct my footage and it does not help the tracking!" If objects still "gloop" together when they shouldn't, then you can try manipulating the Motion Sensitivity parameter. **Motion Sensitivity limits how much pixels can move.** A value of zero assures that pixels can't move very much, and a value of 100 allows pixels to move as much as the motion interpolator can calculate. The default value of 70 might sometimes be a bit too "ambitious" for some material... by reducing the sensitivity you might be able to create a more interesting result (giving a more traditional frame blended look) where there is large motion rather then some sort of inappropriate swirl that **Twixtor** might introduce. Conversely, if there is only one object in the scene and it moves in large amounts, the default value of 70 may not be ambitious enough.

Notes for Interlaced Video Usage Only: Handling Fields

Okay, you ask, what happens when there are fields in the input (interlaced video)?

You can work with interlaced footage and **Twixtor** in one of two ways:

1) Let **Twixtor** create frames from fields using field interpolation. This is accomplished by setting "Fields" in the **Twixtor** settings to "Upper" or "Lower" as appropriate. When passing **Twixtor** footage with fields, each field in an input source becomes a frame to be interpolated. So internally 30i footage really looks like 60p to **Twixtor**. A standard algorithm is applied to make each frame (we interpolate the fields to create a whole frame). Input frame 0.0 is made from the first field (in time, which may be upper or lower depending on your source footage) of the first frame. Frame 0.5 is made from the second field in time of the first frame and so on. Make sure to set Input:Fields to Lower or Upper, as appropriate, in the **Twixtor** settings.

2) You may decide that **Twixtor**'s field interpolation for making frames for a particular piece of footage does not do the job for you, and you would prefer to use some other deinterlacing process and pass **Twixtor** full frames. If so, deinterlace the footage first before applying Twixtor. Then make sure to set Input:Fields to None in the **Twixtor** settings.

NOTE: To see **Twixtor**'s internal deinterlacing, set **Twixtor**'s Time Remap Mode to Speed and set Speed% to 50. Set Input:Fields as appropriate. This will create a frame from each field of your interlaced video.

NOTE 2: Twixtor will perform better if better deinterlacing is performed before the application of **Twixtor**. Our **FieldsKit** plugin set contains such a better deinterlacing plugin, which tries to create better edges, and only performs field interpolation where motion is detected.

NOTE 3: If you place your source footage in a 59.94 fps (NTSC) or 50 fps (PAL) composition (in After Effects or combustion) so that you can view each field as a full frame (or so that you can apply **FieldsKit** to deinterlace each field), then when you apply **Twixtor** make sure to set

Input:Fields to None and the Input:Frame Rate to 59.94 or 50, as appropriate. It's only when you pass Twixtor interlaced material at 29.97 or 25 fps that you should set the Input:Fields to Lower or Upper as appropriate.

For After Effects Only. Handling audio and image sequence time remapping at the same time. "Can I use the host application to time remap both the image sequence and audio, but use **Twixtor** to perform the image calculations, complete with motion vectors?"

The answer to this question is "yes." However, if you want to retime both the audio and video at the same time you won't be able to use Twixtor's Speed% or Frame Num control to control the remapping. You'll have to use the controls that are built in to the host application. Time controls include the "Time Remapping" and "Time Stretch" features of After Effects, and the clip speed and duration settings in Premiere Pro.

Place the material you want to be retimed into a composition (After Effects' term) or sequence (Premiere Pro terms). Place Twixtor on the material and set the Time Remap Mode to Speed and then set Speed% to 100. Set the rest of **Twixtor**'s Track Control and Output Control settings as appropriate. Then nest the composition or sequence (the "Child") into another comp/sequence (the "Parent"). Then retime the Child comp/sequence from inside the Parent comp/sequence using the host's retiming capabilities. That's it!

Note: When working in this way, make sure to turn off frame blending for the host application (if the host application has such a setting).

Note to combustion users: combustion 3 doesn't allow you to time stretch anything but footage, so this technique of remapping audio and image sequences from within combustion is not available.

Conceptually, the hierarchy of control is the following:

- Parent Sequence/Composition
- Retime the child sequence inside the parent sequence or composition using the host application's time remapping controls.
    - Child Sequence (placed inside Parent)
        - Twixtor applied to source material. Time Remap Mode set to Speed, Speed% set to 100.
        - Source material to be retimed (which itself can be another composition or sequence)

PROBLEMATIC FOOTAGE

If you have not used **Twixtor** yet, this section might prove to be a little bit esoteric. This information will be of more use to you as you become more familiar with **Twixtor** and want to understand how to generate the best possible results.

It might happen that **Twixtor** might do poorly in specific instances. The idea here is not to discourage you and tell you it does not work but, rather, to give truth in advertising. As you

become an expert user, you learn to predict the kind of material that can cause problems.

1. "Picket Fence": A very regular pattern in motion (for example someone wearing a t-shirt with fine stripes) with an object moving in front of it (e.g. the same person hand for instance) might confuse the motion vector calculation. Any very structured pattern rotating can cause "blind spots" for the analyzer.

2. "Transparency": Overlay of semi-opaque surfaces might create unexpected results. Some cameras for example will streak under fast motion and that can create disappointing tracking results.

3. "Short Interval Defects" : Sudden global illumination change (e.g., a flash), strobing, dust, ... can create unexpected / undesired results. Also, if there is a piece of hair or a scratch on the scanned film for a frame this would influence the tracking so you really should try to clean such defects before processing.

4. "Fields": We provide some basic field handling. Always remember that by definition, even and odd scan lines in material with fields are considered half a frame apart timewise. Because fields are spatially a line offset from each other, regions where there is no motion and horizontal lines might produce a slight up and down motion in phase with the source frame rate.

5. "Duplicated Frames": **Twixtor** does not provide automatic duplicated frame detection support. As such, you will need to remove duplicated frames before the application of **Twixtor**. Also, you should be aware that if your material has 3:2 pulldown you should remove it beforehand. The same applies to animations on 2's. If you leave the duplicate frames (or fields, in the case of 3:2 pulldown) the freeze-frame, inherent in the duplication of the frames, will be stretched (or sped up, as appropriate).

6. "Alternate Motions": When motions going in different direction are layered it is possible that the dominant motion affects (spills into) the background motion.

7. "Specular Highlights": If you have moving lights, e.g., a shiny object that reflects the light as it moves, it might cause problems because when the motion estimator attempts to match two images, as the motion estimator will tend to follow the highlight as if it was an object. (of course, sometimes this is what you want).

8. "Ultra-Fast Structured Motion": We are very particularly perceptive to human actions. We have sometimes seen that certain complex rapid motion such as someone doing a frenetic dance creates interframe displacements that are just too big for our motion estimator to resolve satisfactorily. When planning a shoot for an effect that involves **Twixtor** in the pipeline, consider that for **Twixtor** fast articulated motion should be easier for front facing subject then sideway views as there will be less pixels traveled per frame on the screen, which is really the only thing that **Twixtor** cares about.

9. "Occlusions": Problems caused by object motions tend to be one frame problems and localized in an area of the frame which is called an occlusion, which is some pixels that you see on one frame but are not visible on the other as a result of camera and/or that object motion.

10. “Limited Reach”: As a rule of thumb considers that **Twixtor** will be most accurate for pixel

displacements that are a maximum of 5% of your image resolution (for 720x486, this comes to a maximum horizontal displacement of 35 pixels or so). With displacements larger than that **Twixtor** will start to become less precise as it tries to separate motions from one another in an image sequence.

11. "Compression Artifacts": Certain video coding techniques such as DV compressors use 8 by 8 pixels blocks (intraframe) based compression. What this means for you is that if for instance you have a sharp edge that moves, it will switch of 8 by 8 block and therefore locally it's neighbor values will be all different (substantially not like 2-3 values over 255 but sometimes 40 off near an edge). This is why green screen like setups perform badly with DV... Without smoothing the source, this can sometimes certainly create tracking problems.

MORE AESTHETIC / COSMETIC CONSIDERATIONS:
Gee! It breaks on these 3 frames what do I do?

Now that we have discouraged you, let's help you to be creative in repairing problems. We know that most of the time, there is nothing we can do to prevent problems at the source because the material already exists, and you need to fix it. So let us share a bit our experience. Experience we acquired playing with this kind of process in production, in trying to make a product that works, and in supporting our users. There is an astronomic quantity of possible images and it's easy to develop assumptions that prove not true if you always work on the same few clips. So, so you know we fully appreciate short movies (5 frames is usually enough) where **Twixtor** fails somehow, this is how we learn, and can hope to make an always better product.

1. 1 or 2 frame defects on source material: Sometimes you receive a movie with one or two frame defects (could be a flash for instance). Take the good frames on both ends of the bad ones and slice them together into another comp, create a short **Twixtor** clip of the proper duration and reinsert it back over the bad frames. Naturally if you also want to apply **Twixtor** to the whole movie, perform the repair before applying **Twixtor** to the whole sequence.

2. Splotches: Although we try to accommodate for separate layers with the Crossing Objects?/Separate Motion? button there are times when you will still get some splotches (unwanted warping artifacts). You can try one of two things:

- In applications that support the animation of pop up menus, you can animate the Motion Vector Quality menu so that it is set to Best for the non-problematic areas and set to None for the problematic areas. For host applications that do not support the animation of pop-up menus (like FCP), you can run **Twixtor** twice (once with Motion Vector Quality set to Best and once with Motion Vector Quality set to No Motion Vectors, for example) and simply replace the problematic images of the Best setting with the images made with the None setting. If your problem only occurs in a small part of the image, you can hand-create a matte to relace the bad portions of the "Best" version with the corresponding parts of the "No Motion Vectors" version, using the compositing functions of your application. You might want to do a soft matte for nice blending. And naturally if you have the **Twixtor Pro** version, you can provide Foreground/Background separation mattes to **Twixtor**, which you should do probably in a manner that respects the shape of the object in the foreground. Additionally, in the **Twixtor Pro** version you can provide guidance to the tracking (see the **Twixtor Pro** manual for more info).

- Try turning down the Motion Sensitivity setting for the problematic portions of the image sequence.

3. Ghosts: Since errors such as ghosting are usually due to occlusions (pixels that were not visible in the previous frame, so don't have correspondence in the other frame due to motion), they are usually easy to paint out so when the result is played in motion no one will see it. Some camera motion such as driving on a bridge tend to create a multi-frame ghost which usually is not too problematic to paint out in apps like Commotion or Combustion that have clone offset paint tools. You might also prefer to not apply motion blur at this stage, paint/fix and then use **ReelSmart Motion Blur** to add some motion blur.

4. Using existing mattes: Sometimes, rotoing out a fast moving foreground object will make the background very stable and produce tracking data without the foreground motion influence. If you have the luxury of having mattes for a layer, and don't have the PRO version, you might want still to try to apply **Twixor** to the foreground and background separately (so that the motion tracking influences on the layers go away or are reduced by Twixtoring them separately). Note in some cases you would need to fill the foreground hole with some form of clean plate image so it has the general motion of it's surrounding area. This process is somewhat what the FG motion masking services of the PRO version would do for you automatically.

5. Shakes...: Although by definition we deal with camera shakes... note that when you slowmo something you will also slowmo the camera shakes and vice-versa, so you might consider to pre-stabilize the sequence on which you want to apply **Twixtor**.

6. Noise, compression artifacts: Our algorithms should not be too sensitive to small grain noise at the analysis level, however it might look weird to extrapolate a lot of frames per frame on material that contains a lot of noise (as it essentially slows down the noise also) so you are advised to remove some noise in the color source, perhaps running a form of smoothing or noise-reduction filter before. With material (such as some "really damaged" MPEG 1 material), lower accuracy motion estimation might even produce more desirable results. Also with DV-like sources, avoid nearest Frame Interp mode. The smoothing of blending actually reduces noise.

7. Very Large SlowMo: The amount of clean frames you can generate is often more a factor of the action in the scene as there is probably no answer to the question of how many frames can be inbetweened with a process like this, as it depends... As a rule of thumb, after 5% of the image size displacement for a single frame, consider yourself lucky if it works. (Again there are no absolute answers). Perhaps also you should consider that there is a limit to how much you slow down things as even perfect inbetweening with a computer animation systems can look weird as it won't have any local dynamics, that is it becomes similar to doing global shape key framing, everything will sort of move at the same rate.

8. Softening: Another related thing to watch out is that you will notice that inbetweens will "soften" the texture details. The thinking is that it's sort of OK as you don't need as much detail on moving elements (you assume you will have some motion blur) and the image where it does not move will remain very clear. If unwanted softening occurs, and the Weighted Blend is not enough, you can consider little post tricks including slightly sharpening (maybe simply an unsharp filter, such as one found in our **SmoothKit** plugin set) and adding a bit of noise... Depending on what you are doing this might be more or less realistic.

9 Handling Multiple Motions: A "rough" garbage matte roto that isolates the foreground object might be necessary on the few frames where the problem occasionally occurs. After applying **Twixtor** to the frames with the garbage matte (on both foreground and background, with inverse mattes of course!), the inbetweened frames can be recomposited back together. When you make a garbage matte for this purpose, you can typically be a bit more "more fat" than the usual roto matte for compositing. In general since Layer selection is not animated it might be simpler sometimes to run the whole thing once and then animate something like the Motion Vector Quality or Motion Sensitivity and then cut back the relevant parts of the two comps together.

Of course, remember that **Twixtor Pro** has many advanced features specifically designed to handle crossing objects and multiple motions.

10. Fast Action: So, with fast action content, you will probably get better results with **Twixtor** from material captured with a 30i FPS camera than with a DV camera that does 12 of 15 real FPS, because the same action will move farther, per frame, in the 12fps footage than in the 30fps footage. If you have the option of shooting progressive or interlaced video, a basic rule of thumb might be if you have fast action (or because you are running with a hand-held camera) shoot interlaced (because we construct 60fps from the 60 fields, which gives us more frames to deal with, which reduces the motion per frame), otherwise shoot progressive (because we don't have to reconstruct a frame from a field, you will get better quality results).

11. Changing Field Order: While **Twixtor** will transform the footage from Upper to Lower field first (or vice-versa), you may want to consider doing the field order conversion first, then applying **Twixtor** with the same input and output field order. The most effective way to change dominance would be to drop the first field, and then reinterlace the fields starting at field 2 of the footage. By doing the field order conversion first you more likely reduce sampling artifacts by **Twixtor**'s creation of whole frames from individual fields. Of course, your mileage may vary :-).

12. "Film Look" and Other Fields Related Problems: If Fields cause you problems, remember that we offer also a full **FieldsKit** which allows to flexibly organize your workflow to possibly create a better deinterlaced source for **Twixtor**. Note that some people want to simulate a telecine look by going down to 24 fps and then adding a 3:2 pulldown. This is certainly possible and **FieldsKit** will eliminate the need to create an intermediary render. But, it is our experience that you obtain something closer to a film look by simply Deinterlacing and applying ReelSmart Motion Blur and in that case maybe you don't need **Twixtor**. For more on our products and "film look" issues, please read our film look FAQ (located here).

13. Flickering: Similarly weird flickering as sometimes happen with problematic film capture can have an impact. Note **Twixtor** mostly uses the black and white information and is by default hardwired to video brightness, which largely discounts blue. Sometimes just doing an RGB invert of the color source and using it as the tracking source will help. So particularly for shots that have a lot of blue, it might be worth to supply an alternate track which remaps colors and pushes up certain values. This also applies with certain dark shots (low-lights level), it might be worth to expand their range and lift them up (making sure not to clamp values that would make entire regions "look the same" to **Twixtor**'s motion estimator).

"Okay, I have material that still needs more work. What can I do to make it look even better?"

Remember that **Twixtor Pro** has many advanced features specifically designed to help you fix problematic

footage in a more intuitive and direct manner.

Twixtor, ReelSmart, FieldsKit and SmoothKit are trademarks of RE:Vision Effects, Inc.
Adobe, After Effects, and Premiere are registered trademarks of Adobe Systems, Inc., Combustion is a trademark of Autodesk, Inc.

Copyright 2000-2010 RE:Vision Effects, Inc.